WWW.SCIAM-MAGAZINE.NAROD.RU

дующем номере я приведу найденное мною общее решение.

Вы можете поиграть с системами зеркал, образующими в сечении равносторонний многоугольник с четным числом сторон. Некоторые из создаваемых ими полей изображений будут напоминать картинки, создаваемые двумя параллельными зеркалами. Есть ли между ними различие? Что будет, если взять многоугольник с нечетным числом сторон? Что вы увидете, если заглянете внутрь калейдоскопа с отражающей цилиндрической поверхностью, которая есть не что иное как многоугольник с бесконечным числом сторон? Думаете ли вы, что калейдоскоп с кривыми зеркалами должен создавать однозначное поле?

Издательство **МИР** *предлагает:*

С. Кейслер

ПРОЕКТИРОВАНИЕ ОПЕРАЦИОННЫХ СИСТЕМ ДЛЯ МАЛЫХ ЭВМ

Перевод с английского

Рассматриваются методы проектирования операционных систем многоуровневого управления малыми вычислительными машинами, предназначенными для использования в автоматизированных комплексах управления производственными процессами или в качестве персональных компьютеров. Описываются основные блоки операционной системы, указывается их назначение. Большое внимание уделено взаимодействию операционных систем с пользователем. Изложение сопровождается многочисленными примерами.

Для специалистов, работающих в области создания программного обеспечения мини- и микро-ЭВМ.

1986, 39 л. Цена 3 р. 10 к.

Занимательный компьютер

О линейках Коломба и их применении в радиоастрономии

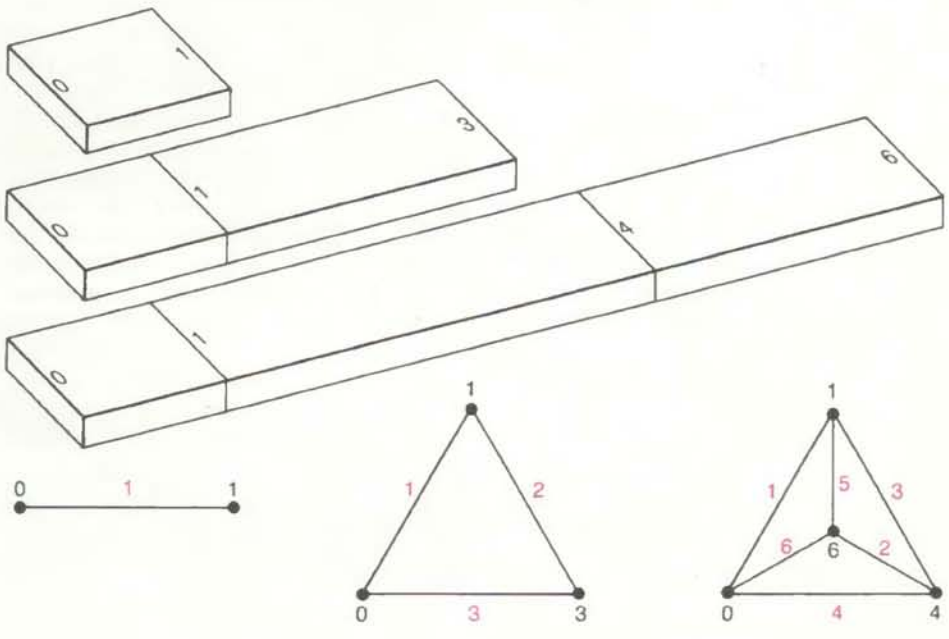
А. К. ДЬЮДНИ

ПРИ ПОМОЩИ простой линейки длиной в один фут с 13 делениями, каждое из которых соответствует одному дюйму, можно измерить 12 различных расстояний. Возможно ли так усовершенствовать это знакомое всем устройство, чтобы число измеряемых с его помощью расстояний было больше числа нанесенных на нем делений? Оказывается, да: например, на стандартной линейке можно оставить лишь пять делений и при этом измерить 10 различных расстояний. Каждое такое расстояние заключено между некоторой парой делений и вычисляется как разность между целыми числами, которыми помечены эти деления. Того же результата можно достичь, даже воспользовавшись 11-дюймовой линейкой. Читатели, которые после некоторых размышлений придумают, как это можно сделать, создадут тем самым так называемую линейку Коломба.

Поиск таких линеек — весьма интересное занятие, в котором большую помощь может оказать компьютер. Решая эту задачу, вы не просто удовлетворяете свое любопытство, а участвуете в серьезном исследовании. Дело в том, что необходимость в линейках Коломба возникает в целом ряде областей науки и техники.

Эти устройства являются изобретением Соломона У. Коломба, профессора математики и электроники Южнокалифорнийского университета. В течение двух десятилетий он и его коллеги занимались поиском и изучением свойств этих линеек. Устройства подобного рода могут применяться в теории кодирования, рентгеновской кристаллографии, конструировании электронных схем, радиоастрономии.

Одним из исследователей, в работе которых линейки Коломба играют важную роль, является Д. Робертсон, геофизик, работающий в Роквилле (шт. Мэриленд) в Национальном геодезическом центре Национальной службы США по океану и атмосфере. Он применяет радиоастрономический метод — интерферометрию с очень длинной базой — не для того, чтобы фиксировать местоположение радиоисточников, а чтобы выполнить точные измерения параметров Земли. Занимаясь уже в течение нескольких лет



Три идеальные линейки Коломба и диаграммы их расстояний

WWW.SCIAM-MAGAZINE.NAROD.RU

	ЧИСЛО ДЕЛЕНИЙ	КРАТЧАЙШАЯ ИЗВЕСТНАЯ ЛИНЕЙКА	НИЖНЯЯ ГРАНИЦА
ИДЕАЛЬНАЯ ЗОНА	2	1	
	3	3	
	4	6	
ЗОНА ЗНАНИЯ	5	11	
	6	17	
	7	25	
	8	34	
	9	44	
	10	55	
	11	72	
	12	85	
	13	106	
ЗОНА ИССЛЕДОВАНИЙ	14	127	114
	15	155	133
	16	179	154
	17	199	177
	18	216	201
	19	246	227
	20	283	254
	21	333	283
	22	358	314
	23	372	346
	24	425	380

Длина линеек Коломба и линеек, близких к этому классу

поиском линеек, он обращается за помощью к читателям. В результате, возможно, удастся точнее определить размеры, форму и параметры движения нашей планеты.

Прежде чем откликнуться на просьбу Робертсона, нам, наверное, следует вникнуть в принципы, лежащие в основе линеек Коломба. Хотя линейки могут быть произвольной длины, пока изучены лишь линейки малых размеров. Первые три линейки можно описать несколько абстрактно тремя последовательностями чисел:

0, 1
0, 1, 3
0, 1, 4, 6

Чтобы представить себе линейку как физический объект, можно мысленно пометить левый конец «пустой» линейки длиной n единиц наименьшим числом последовательности, т. е. 0. Правый конец пометим наибольшим числом n; им может быть 1, 3 или 6. Промежуточными числами пометим деления, нанесенные на соответствующих расстояниях от левого конца линейки, т. е. от 0 (см. рисунок на с. 103).

Простой способ проверить, какие расстояния из диапазона от 0 до n можно измерить одной из этих маленьких линеек, заключается в том, чтобы нарисовать диаграмму расстояний на линейке. Для этого каждому целому числу на линейке поставим в соответствие точку на листе бумаги и пометим точку данным числом. Затем каждую пару точек соединим линией, помеченной разностью между целыми числами, расположенными на концах линии. Если каждое расстояние в диапазоне от 0 до n появляется на одной и только на одной линии диаграммы расстояний, то линейка Коломба называется идеальной. Все три линейки на иллюстрации идеальные — факт, в котором можно убедиться, взглянув на соответствующие диаграммы расстояний. На каждой диаграмме ни одно расстояние не появляется более одного раза, и все возможные расстояния в диапазоне от 0 до n присутствуют.

Среди линеек Коломба идеальные встречаются очень редко. На самом деле все существующие идеальные линейки были только что рассмотрены. Для значений n больших 6 «неидеальность» может проявляться двояким образом: либо какое-то расстояние встречается более одного раза, либо не встречается вовсе.

Вот так, в самом начале поиска бóльших (и лучших) линеек мы столкнулись с суровой действительностью. Однако откуда известно, что идеальных линеек бóльших размеров не существует? Коломб представил короткое и изящное доказательство этого факта.

Он предлагает рассмотреть не сами деления на линейке, а интервалы между ними. Если линейка идеальна, то множество всех интервалов между последовательными делениями образует ряд 1, 2, 3... до $m - 1$ в некотором порядке (m — количество делений). Коломб спрашивает: «Где расположен единичный интервал?». Если он прилегает к любому интервалу, длина которого меньше $m - 1$, то оба эти интервала вместе дают расстояние, меньшее m. Но это расстояние уже должно было встретиться в виде одного из интервалов где-то еще, потому что все расстояния от 1 до $m - 1$ расположены соответственно между последовательными делениями. Указанные соображения приводят нас к несколько неожиданному выводу, что единичный интервал должен прилегать к интервалу длиной $m - 1$. Более того, по другую сторону от единичного интервала никаких интервалов больше нет, ибо он располагается на одном из концов линейки.

Полученный нами результат представляет собой как бы один поворот ручки того, что математики называют «колбасной машиной». Далее, Коломб спрашивает: «Где располагается интервал длиной 2 единицы?» Повернем ручку еще раз. Если интервал длиной 2 единицы прилегает к любому интервалу длиной меньше $m - 2$, то взятые вместе эти два интервала образуют расстояние, уже встречавшееся где-то еще. На этот раз мы уже не можем считать, что интервал длиной 2 единицы прилегает к интервалу длиной $m - 2$. Их суммарная длина m уже измеряется комбинацией интервалов длиной 1 и $m - 1$ с одного конца линейки.

Таким образом, колбасная машина останавливается, и мы приходим к выводу, что есть только один интервал, к которому может прилегать интервал длиной 2 единицы, а именно интервал длиной $m - 1$. Поскольку у любой линейки есть только два конца, идеальная линейка не может содержать больше трех интервалов: 1, $m - 1$ и 2. Доказательство завершается, когда мы вспоминаем, что три интервала требуют четырех делений. Таким образом, при $m = 4$ интервалы равны 1, 3, 2, а соответствующие деления образуют последовательность 0, 1, 4, 6.

Столкнувшись с полным отсутствием идеальных линеек, имеющих более 4 делений, математик попытается «отыграться», построив новое определение. Продемонстрируем пример определения, возникшего в результате подобных попыток. Наилучшим приближением к невозможной идеальной линейке с пятью делениями является такая линейка, которая содержит каждое расстояние не более одного раза, но может и не содержать всех расстояний, которые должна была бы иметь идеальная линейка того же размера. Этому условию легко удовлетворить, подобрав достаточно длинную линейку. Среди всевозможных линеек с пятью делениями, содержащих каждое расстояние максимум по одному разу, выявим самую короткую. Такая линейка будет называться линейкой Коломба пятого порядка. Линейка Коломба порядка m определяется аналогичным образом. Поскольку это определение включает также первые три линейки как частный случай, оно как бы сглаживает разрыв в свойстве «идеальности», наблюдающийся при переходе через значение m, равное 4.

Г. Тэйлор, коллега Коломба, представил имеющуюся на данный момент информацию о линейках Коломба в виде приведенной вверху таблицы. В диапазоне от двух до 24 делений мы располагаем как точными знаниями, так и некоторыми гипотезами относительно размеров линеек Коломба. Зона, которую я называю идеальной, заключается между двумя и четырьмя делениями. Далее прости-

WWW.SCIAM-MAGAZINE.NAROD.RU

рается зона знания, охватывающая линейки Коломба, имеющие до 13 делений. В этой области известны все линейки минимального размера. Другими словами, в каждом конкретном случае не существует линейки Коломба меньшей длины с тем же количеством делений. Линейка Коломба с 5 делениями имеет длину 11 единиц. Линейка Коломба с 13 делениями имеет длину 106 единиц.

За 13 делениями начинается зона, которую я называю зоной полумрака. Эта зона (назовем ее более уважительно зоной исследований) содержит только те линейки, относительно которых еще нельзя с уверенностью сказать, что они являются линейками Коломба. Для каждого количества делений в таблице указана длина линейки. Однако могут существовать и линейки меньшего размера. На самом деле имеется формула, которая дает нижний предел для этих размеров. Постоянно увеличивающийся разрыв между значениями, получаемыми по формуле, и длинами найденных к настоящему моменту линеек свидетельствует либо о слабости оценки по формуле, либо о том, что линейки, найденные для каждого значения числа делений, имеют тенденцию «ухудшаться» с ростом последнего.

Зона знания была расширена до границы в 13 делений благодаря усилиям Робертсона. В ходе непрерывной работы ЭВМ в течение месяца его программа произвела исчерпывающий перебор всех потенциальных линеек Коломба с 13 делениями и нашла самую короткую. Если для поиска линеек Коломба с 14 и более делениями потребуется еще больше машинного времени, то, наверное, немногие читатели заинтересуются этим поиском.

Вместо этого можно предложить им тактику «компьютерных налетов» на зону исследований, в результате которых можно получить линейки, обладающие меньшими размерами, чем найденные до сих пор. Очень полезной для всех поисков подобного рода является программа под названием CHECKER. Однако прежде чем применить эту программу, необходимо задать способ проверки наличия одинаковых разностей между всеми возможными парами целых чисел в данном массиве. При самом простом подходе можно организовать массив таких разностей, а затем просмотреть его в поисках дублирующихся значений, опираясь на довольно неуклюжий и требующий много времени алгоритм.

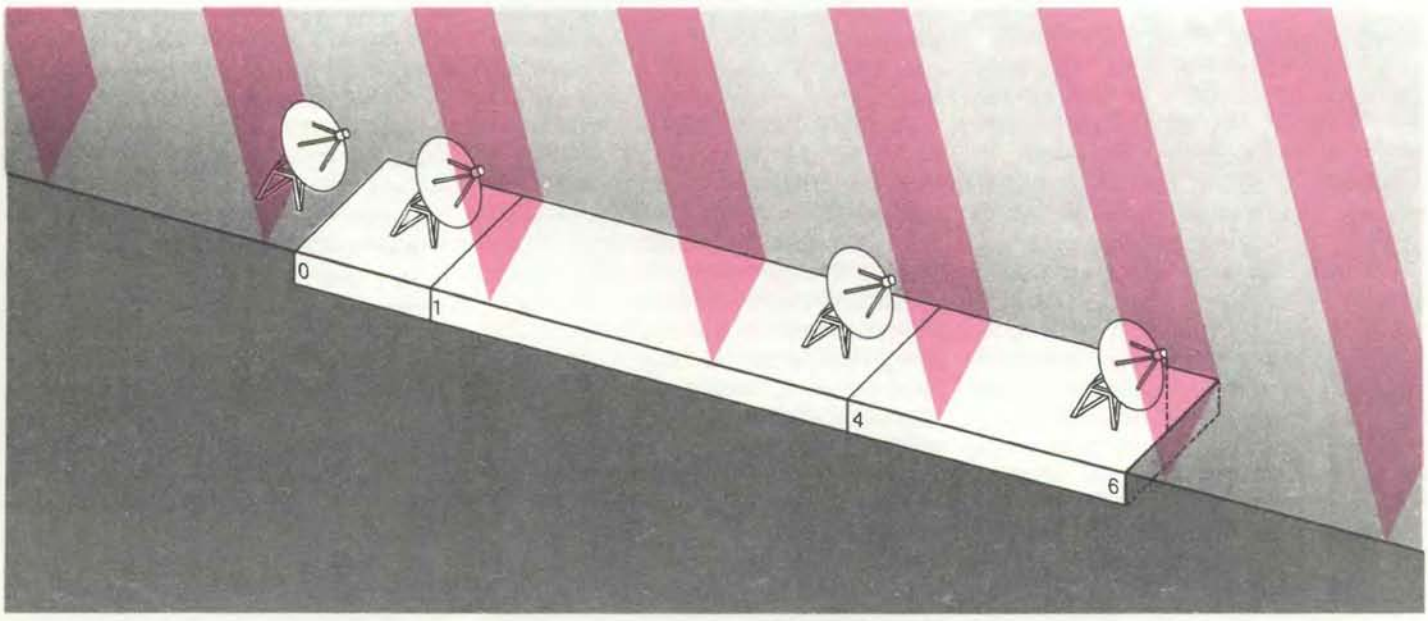
Редко бывает так, чтобы алгоритм, обеспечивающий наиболее быстрое решение задачи, можно было реализовать с помощью самой короткой программы. Однако здесь мы сталкиваемся как раз с таким случаем. Поскольку предполагается, что все разности отличаются друг от друга (т. е. одинаковых разностей не существует), то ими можно воспользоваться как адресами в специальном массиве *check*. Вначале в массиве *check* хранятся только нули. Каждый раз, когда вычисляется новая разность, значение, хранящееся по соответствующему адресу, меняется с 0 на 1. Таким образом, по мере того, как программа CHECKER проводит дальнейшие вычисления, она может наткнуться на единицу, уже записанную по данному адресу, а это означает, что «новая» разность уже встречалась где-то раньше. В этом случае рассматриваемая линейка не может принадлежать к классу Коломба, поскольку она не обладает основным свойством, присущим линейкам Коломба: каждое расстояние должно встречаться не более одного раза.

Описанный метод, использующий значения разностей в качестве адресов, представляет собой простую разновидность того, что специалисты по вычислительной математике и программированию называют хешированием. Во многих задачах, связанных с информационным поиском, хеширование обеспечивает самый быстрый доступ к данным.

Для тех, кто хотел бы подробнее ознакомиться с программой CHECKER, опишем в общих чертах ее структуру. С помощью двух вложенных циклов генерируются все возможные пары целых чисел из входного массива. В первом цикле генерируются индексы i, а во втором — индексы j. Программа вычисляет абсолютное значение разности и присваивает его в качестве значения переменной *diff*. На следующем шаге алгоритма программа CHECKER использует значение *diff* в качестве хеш-кода. В алгоритмических обозначениях можно записать:

если *check*(*diff*) = 1
  то вывести «не коломбова» и выход
  в противном случае *check*(*diff*) ← 1

Если программа ни разу не выведет сообщение «не коломбова», то мож-



Пары радиотелескопических антенн, расположенных по линейке Коломба, позволяют определить разность фаз приходящих сигналов

WWW.SCIAM-MAGAZINE.NAROD.RU

но считать, что рассматриваемая линейка успешно прошла главный тест. Однако каков ее размер? Существует несколько способов получить ответ на этот вопрос.

Прежде всего можно воспользоваться программой CHECKER в ее основном варианте без каких-либо дополнительных программ. Я представляю себе читателя, склонившегося над клавиатурой и увлеченно работающего с программой CHECKER. Он проводит разведку в зоне исследований на высоте 14 делений в поисках линейки короче 127 единиц — именно такую длину имеет лучшая из известных на сегодня линеек. Читатель, совершающий полет в сложных метеорологических условиях, не знает, в какую сторону повернуть. Только что он ввел последовательность из 14 делений. Наибольшим числом этой последовательности было 124. С неописуемым волнением он видит, как на экране начинает высвечиваться сообщение: «Поздравляю! Последовательность подходит». Введя это поздравительное сообщение в программу, читатель дает себе слово всегда пробовать ее исключительно для потенциально рекордных последовательностей.

Возможно, читатель нашел свой рекордный набор делений, последовав совету Коломба и исследуя только те линейки, у которых наиболее длинные интервалы встречаются в средней части. У таких линеек интервалы уменьшаются при приближении к концам, однако точные закономерности читателям предстоит найти самим. Коломб заверяет нас, что многие хорошие линейки, хотя и не обязательно наилучшие, обладают этим свойством.

Программу CHECKER можно несколько модифицировать, чтобы придать изысканиям характер коротких пробных попыток. В программе STEP CHECKER целые числа вводятся по одному. После того как введено очередное число, программа вычисляет все разности между только что введенным числом и теми, которые уже хранились в памяти. На самом деле STEP CHECKER — это просто версия программы CHECKER, в которой инструкция ввода заменяет внешний цикл. Последовательность считается подходящей, если при обработке последнего введенного числа программа не выдала сообщения «не коломбова».

Наконец, программу STEP CHECKER можно встроить в автоматическую систему, подобную разработанной Робертсоном. Его программа, которую я могу назвать EXHAUST (от англ. exhaustive — исчерпывающий), поскольку она производит исчерпывающий перебор, систематически порождая новые линейки путем добавления новых интервалов. После добавления каждого нового интервала программа STEP CHECKER проверяет полученную таким образом последовательность, или линейку.

Робертсон построил свою программу исходя из того, что новые интервалы (и соответственно новые деления) должны добавляться к линейке слева направо. Читатели, которые следили за ходом доказательства Коломба (о несуществовании идеальных линеек для числа делений больше 4), вспомнят, что интервалы должны иметь длину $1, 2, \ldots, m-1$ в некотором порядке. Хотя это было справедливо только в отношении идеальных линеек, нечто подобное, вообще говоря, справедливо также в отношении линеек Коломба. Большинство, хотя и не все, интервалов длиной от 1 до $m-1$ между последовательными делениями на линейках, приближающихся к идеальным, встречается в некотором порядке. Отметим, однако, что у таких линеек встречаются интервалы даже больше $m-1$.

Алгоритм Робертсона порождает новые интервалы в пошаговом режиме. Эти интервалы хранятся в массиве, который мы назовем *spaces*. Программа EXHAUST просматривает массив и добавляет к нему каждый раз по одному новому интервалу. Конечно, в алгоритме используются некоторые простые тесты, облегчающие работу программы. С помощью одного из них можно проследить, чтобы новый интервал не совпадал с одним из уже имеющихся в массиве *spaces*, а с помощью второго — чтобы суммарная длина всех интервалов, образующих линейку, не превосходила минимальную известную длину для линейки с данным числом делений.

Программа EXHAUST, кажется, с большим энтузиазмом отыскивает новые линейки. Сначала она полагает первый элемент массива *spaces* равным 1 и добавляет по единице к следующему интервалу, так чтобы он отличался от первого. Затем она начинает добавлять по единице к третьему интервалу, следя за тем, чтобы это расстояние не только отличалось от первых двух, но также удовлетворяло условию, заложенному в программе STEP CHECKER, согласно которому все расстояния, содержащиеся в линейке, должны отличаться друг от друга. Всякий раз, когда появляется новый элемент (интервал), программа суммирует все элементы массива и сравнивает полученную сумму с длиной самой короткой известной к данному моменту линейки. Если сумма оказывается меньше, программа переходит к следующему интервалу. Если нет, то она возвращается к предыдущему элементу и продолжает перебор.

Программа Робертсона работает значительно быстрее, если первый элемент массива *spaces* положить равным 2, а не 1. На самом деле прогон программы, занимающий целые сутки, сократится на несколько часов. Читатели могут поразмыслить над тем, почему и в этом случае поиск остается исчерпывающим.

Конечно, эффективность алгоритма поиска можно повысить за счет включения других тестов и эвристических приемов. Эти тесты и эвристики могут особенно сильно повлиять на эффективность, если они налагают дополнительные ограничения на возможные значения элементов массива *spaces*. Возможно, существует процедура, работающая со значительно более узкими диапазонами этих значений. Обработка массива во многом подобна счету. А счет будет выполнен намного быстрее, если уменьшить количество возможных вариантов в каждом элементе массива *spaces*. В любом случае читатели, сумевшие найти рекордного бобра-работягу, построить новые катапульты, стреляющие планерами, и добившиеся многих других результатов в своих исследованиях, смогут найти свой собственный подход к решению задач из зоны исследований Коломба. О лучших находках будет рассказано в одной из наших следующих статей. Интересно, кто первым найдет две различные линейки, имеющие больше 6 делений, но измеряющие один и тот же набор отличных друг от друга расстояний. Линейки, представляющие собой взаимные зеркальные отображения, не считаются различными.

Успешное решение задачи будет означать несостоятельность «теоремы», предложенной швейцарским математиком Софи Пиккард в 1939 г. Согласно этой теореме, две линейки, измеряющие один и тот же набор отличных друг от друга расстояний, должны быть идентичными. Теорема очень понравилась специалистам по рентгеновской кристаллографии, поскольку она помогала им разрешать неопределенности в дифракционных картинках. К сожалению, эта теорема оказалась не справедливой для многих пар линеек, имеющих 6 делений. Может быть она остается справедливой для всех линеек более высокого порядка. Этот вопрос можно рассмотреть, выбрав в качестве объекта исследований линейки, имеющие всего 7 делений.

WWW.SCIAM-MAGAZINE.NARODU.RU

Однако какое это все имеет отношение к задаче, которой занимается Робертсон? Дело в том, что линейка Коломба иногда используется в радиоастрономии для разрешения удаленных радиоисточников и для измерения параметров нашей планеты. В первом случае несколько антенн располагаются на прямой линии протяженностью несколько километров. Позиции антенн соответствуют делениям на линейке Коломба (см. рисунок на с. 105). Для того чтобы определить положение удаленного радиоисточника, необходимо найти угол между линией базы антенн и направлением волнового фронта, поступающего от источника. Все антенны настроены на одну и ту же длину волны. Точное время прибытия сигнала к каждой антенне может быть установлено путем анализа записи принимаемых сигналов. Полное число длин волны, укладывающееся в интервале между данной парой антенн, называется полной разностью фаз. Обычно она состоит из целого числа и дробной части, которая называется сдвигом по фазе. Если удается восстановить полную разность фаз, то искомый угол между направлением на источник и линией базы нетрудно вычислить по известной длине волны и скорости света c. Однако по сигналам, принятым каждой парой антенн, можно определить лишь сдвиг по фазе, но не полную разность фаз.

В действительности полная разность фаз вычисляется с применением анализа Фурье по данным, зафиксированным многими парами антенн. Однако если расстояние между одной парой антенн оказывается равным расстоянию между какой-нибудь другой парой, то обе они дают одну и ту же информацию о сдвиге по фазе. Избыточность информации означает ее потерю. Точность вычисления искомого угла наиболее высока в том случае, когда каждая пара антенн фиксирует отличный от других сдвиг по фазе. Это условие выполняется за счет размещения антенн по тому же принципу, по которому располагаются деления на линейке Коломба.

Еще один способ определения положения удаленного радиоисточника заключается в том, чтобы воспользоваться лишь двумя приемниками, сканирующими одновременно весь набор длин волн, на которых ведутся наблюдения. Прием на две антенны на нескольких различных волнах позволяет получить ту же информацию о полной разности фаз между принимаемыми сигналами, какую дало бы использование нескольких антенн, настроенных на одну и ту же длину волны. Здесь опять возникает опасность появления избыточной информации. Ни одна пара длин волн, на которых ведется наблюдение, не должна иметь разность, совпадающую с разностью между какой-либо другой парой. И хотя линейка Коломба здесь остается как бы невидимой, она все же присутствует в неявной форме.

Робертсон применяет второй метод не для фиксации радиоисточников, а для того, чтобы определить местоположение самих антенн. Для целей, которые он преследует, недостаточно знать, что вторая антенна находится в Уэстфорде (шт. Массачусетс). Ему нужно знать ее положение с точностью до нескольких сантиметров. Такой точности можно достичь, если воспользоваться очень далеким, практически точечным источником, таким, как квазар. Установить точное положение точки на земной поверхности по отношению к далекому радиоисточнику равносильно тому, чтобы точно установить значения таких фундаментальных земных параметров, как диаметр планеты, ориентация оси ее вращения и продолжительность суток. На уровне сантиметров и микросекунд эти параметры являются переменными величинами из-за воздействия годовых, сезонных и даже метеорологических вариаций, которые иногда подчиняются известным закономерностям, а иногда остаются совершенно необъяснимыми.

В НОЯБРЬСКОМ номере нашего журнала была описана доброкачественная терминальная болезнь КРАБЫ. Орды морских ракообразных внезапно, без предупреждения, с жадностью набрасываются на терминалы Блит, за которыми работают сотрудники фирмы AT&T Bell Laboratories в Мюррей-Хилле (шт. Нью-Джерси).

Почти все программы, выполняющиеся в мультипрограммной системе терминалов типа Блит, пользуются важным оператором *bitblt*. С помощью этой операции прямоугольные множества битов переносятся из одной области памяти терминала в другую. Перед выполнением переноса прямоугольные множества могут подвергаться различным логическим операциям.

Перед читателями были поставлены две задачи: стереть изображение на экране и повернуть изображение на 90°. Первая задача решается довольно легко. Чтобы стереть картинку, занимающую определенное прямоугольное множество, нужно применить операцию «исключающего или», взяв в качестве операндов само множество. В результате получится множество, содержащее только нули. Таким образом, при помощи одной-единственной операции *bitblt* можно полностью очистить заданную область экрана.

Повернуть изображение значительно труднее, чем стереть его. До сих пор лишь один читатель представил приемлемое решение. Старшеклассник Т. Вителски из Нью-Йорка нашел способ повернуть изображение размером $n \times n$ на экране размером $2n \times 2n$. Затратив лишь $3n - 1$ операций *bitblt*, он проходит через $n - 1$ строк, копирует подызображение размером $(n - 1) \times (n - 1)$, проходит через $n - 1$ столбцов, копирует другое подызображение и затем проходит еще через $n - 1$ строк. Метод Вителски оказался более быстрым, чем стандартный алгоритм поворота, требующий $4n - 2$ операций.

Читатели, не чувствующие себя достаточно компетентными в программировании, чтобы написать свои собственные версии программы MANDELZOOM, но желающие завести ее в библиотеке программ своих персональных компьютеров, обратились к нам с просьбами прислать им программу. Когда читатели просят прислать им листинги или дискеты с записью программ, которые я представляю в наших статьях, возникает, конечно, соблазн удовлетворить их просьбу. Но у меня просто не хватит времени, чтобы ответить на все поступающие письма. Кроме того, я считаю, что время, которое читатель затратит, чтобы научиться нескольким простейшим элементам программирования, требующимся для решения большинства описываемых мною задач, он проведет с пользой для себя. С педагогической точки зрения это, наверное, правильно.

В то же время, когда другие читатели щедро предлагают поделиться созданными ими программами со всеми желающими, я уже не могу удержаться от соблазна. Поэтому иногда я привожу имена любителей, выразивших такое желание.